EFREI
M1 , 2015-2016

T.P.1 SCV TINA ToolBox

Introduction

The TINA (TIme petri Net Analyzer) toolbox (http ://www.laas.fr/tina/) supplies several tools allowing to edit, to simulate and to analyze Petri nets. The objective of this tutorial is to familiarize yourself with these tools. Feel free to read the documentation and test the editor on simple examples before processing the following exercises.

Exercice 1 : Mutual Exclusion Algorithm

Consider the the mutual exclusion algorithm seen in the lecture.

1. Draw the corresponding Petri net.
2. Simulate the behavior step by step (onglet `tools/stepper`).
3. Simulate a behavior leading to a deadlock state.
4. Build the corresponding reachability marking graph and analyse the generated file.
5. Express (using the LTL logic) and check the fairness and the deadlock freeness properties.

Exercice 2 : Traffic light

1. Model with a Petri net the acyclic activity of ta three colors traffic light.
2. Represent the behavior of two trafic lights coordinating the traffic at the intersection of two roads. A traffic light switch to the green color only when the other light is red. Initially one is red while the other is green.
3. Draw the corresponding reachability marking graph.
4. Express (using the LTL logic) and check the following properties (in case the property is not satisfied analyse the supplied counterexample) :
   (a) The two lights can never be both green at the same time.
   (b) When a light is red, then it will become green in the future.

Exercice 3 : Lecteurs Ecrivains

The following Figure illustrates a Petri net model of a concurrent access to a data base problem. We consider 6 readers and 3 righters. In order to ensure the coherence of the data base, one must prohibit the access (to read or to write) to all the processes when a process is modifying the database. However, several simultaneous reading accesses are possible.

1. Use the TINA editor to draw the Petri net.
2. Build the reachability marking graph and give the number of states and the number of edges.
3. Express (using the LTL logic) and check the following properties (in case the property is not satisfied analyse the supplied counterexample) :
   (a) Place `lock` is 5-bounded.
   (b) Reading and writing in the database are mutually exclusive.
   (c) At most, 5 five reader can read simultaneously in the database.
   (d) Any `reader` arrival will be followed a `read`.
   (e) Any reading start will be followed by a reading end.

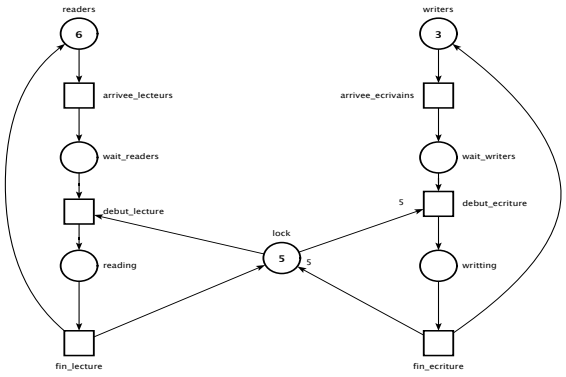

FIGURE 1 – Concurrent Access to a database

Exercice 3 : The barber problem

Model with a Petri net the barber problem while according to the following specification :

— The capacity of the waiting room of the hairdresser saloon is n,

— The barber can be either in the state `free` or in the state `busy`,

— When a client enters the saloon, he/she can have one of the three possible behaviors :

  — If the barber is free and the waiting room is empty, he/she can be served immediately. The barber then enters the state `busy`,

  — If there is an empty chair in the waiting room, he/she takes a chair and waits for his/her turn,

  — If the waiting room is full, the client leaves the saloon.

In the following, we consider that $n = 2$.

1. Start the simulator and verify that your model satisfy the specification.
2. Is the Petri net bounded ? If no, simulate a behavior that illustrates the non boundedness of the model.
3. Build the coverability marking graph of the net. Analyse the generated file. What is the meaning of the ω symbol.
4. Change the Petri net to make it bounded.
5. Build the reachability marking graph of the obtained model.
6. Imagine, express and check some properties (at least three) that can be meaningful according to the specification of the system.